Ville Kemppainen

Correspondence analysis as a service

Helsinki Metropolia University of Applied Sciences

Bachelor of Engineering

Media Engineering

Thesis

18 September 2017

Helsinki Metropolia University of Applied Sciences

Author Title Number of Pages Date	Ville Kemppainen Correspondence analysis as a service 35 pages + 1 appendix 31 August 2017
Degree	Bachelor of engineering
Degree Programme	Media engineering
Instructor	Olli Alm, senior lecturer

Correspondence analysis is examined and demonstrated in this thesis from the viewpoint of service development. Traditionally correspondence analysis has been manual, the procedure varying due to source data formats varying. However, automating the process from recognising suitable data to making insights from it, is possible and applicable to a variety of fields. This motivates exploring its service viability.

The use of correspondence analysis is to discover and visualize corresponding categorical variables. A prototype for analysing a case example dataset was built. Also the statistical quality of the data was evaluated. The example data are the results to a survey conducted by The Finnish Innovation Fund Sitra. The pairs of survey questions were visualised.

High statistical significance and independent association in a notable share of the survey data was found. Through general scrutiny of correspondence analysis and the data provider's feedback about the example implementation, it was found that correspondence analysis visualisations can be interpreted moderately easily.

Implementing data analysis techniques as service components is challenging but viable. Extending the analysis technique beyond the classical definition is important to the viability of correspondence analysis. To improve interpretability, complementary information should be produced through Principal Component Analysis, cluster analysis and classification algorithms.

Keywords	data analysis, data visualisation, software-as-a-service, categorical data

Tekijä Otsikko Sivumäärä Päiväys	Ville Kemppainen Korrespondenssianalyysi palveluna 35 sivua + 1 liite 31.8.2017
Tutkinto	Mediatekniikan insinööri
Koulutusohjelma	Media Engineering
Ohjaaja	Lehtori Olli Alm

Insinöörityössä tutkittiin korrespondenssianalyysin soveltuvuutta palveluksi analysoimalla kyselydataa. Korrespondenssianalyysi on data-analyysitekniikka, jolla voidaan löytää ja visualisoida toisiaan vastaavia kategorisia muuttujia. Korrespondenssianalyysiprosessin automatisointi datansyöttövaiheesta tulkintavaiheeseen on mahdollista, ja muun muassa markkinointiautomaatio- ja tutkimussovellukset tuottavat kategorista dataa.

Ohjelmistopalvelukokonaisuuden osaksi soveltuva korrespondenssianalyysin toteutus rakennettiin JavaScript-moduuleista. Esimerkkidatana toimi Suomen itsenäisyyden juhlarahasto Sitran tuottaman kyselyn vastausdata. Esimerkkidatan konversioita ja tilastollista varmentamista varten ohjelmoitiin uudet moduulit. Kaikkien monivalintakysymysparien välinen korrespondenssi visualisoitiin.

Selvisi, että 5000 vastaajan tuottama esimerkkidata on hyvälaatuista. 82 kysymyksestä muodostetuista pareista useimmat ovat tilastollisesti merkittäviä ja niiden korrespondenssi on riippumatonta. Korrespondenssianalyysin ominaisuuksien tutkimisesta ja esimerkkitoteutuksen testaustuloksista käy ilmi, että korrespondenssianalyysin tuloksia voi tulkita ilman laajaa data-analyysin erikoisosaamista.

Data-analyysitekniikoiden hyödyntäminen palvelukokonaisuuksina on haastavaa. Korrespondenssianalyysin tapauksessa analyysitekniikan laajentaminen on palvelun toimivuuden kannalta tärkeää. Tulkittavuuden parantamiseksi käyttäjälle tulee antaa sanallista ja kuvallista lisätietoa. Lisätietoa voidaan tuottaa esimerkiksi pääkomponenttianalyysilla, klusterianalyysilla ja luokittelualgoritmeilla.

Avainsanat	data-analyysi, datan visualisointi, ohjelmisto palveluna, kategorinen data

Contents

1 Introduction

People are increasingly data-literate (Cairo 2016). A complementing trend is that the market for data analysis software services is growing. This is because data analysis, the practice of deriving insights from data, can be used extensively to confirm business decisions and to aid business operations. In order to implement data-aided business operations, businesses begin with collecting data, using various software services. Web analytics services like Google Analytics (Google Analytics Solutions 2017) provide web traffic metrics and information about user behaviour. Similarly, Customer Relationship Management software is used to collect data in order to better manage the complexity of business leads and to differentiate between active and idle customers.

However, collecting data does not guarantee insights. Reading data can be challenging in many ways. Practical issues that can make data problematic are the lack or vagueness a research plan, metadata and/or documentation. Challenges can also be theoretical – for instance, the challenge that correspondence analysis solves is the high dimensionality of categorical data. Inexpensive automated analysis can create business incentive for challenging data that has no incentive for manual analysis. Software systems can, for instance, automatically apply statistical techniques and visualise the results in order to communicate insights. A generic example of this are the visualisations produced by Google Analytics (Google Analytics Solutions 2017).

Correspondence analysis, a graphical data analysis technique aimed for particularly multivariate categorical data, is described and demonstrated in this thesis from the viewpoint of software service development. Correspondence analysis is a statistical technique that visualizes the correspondence of multiple categorical variables in a 2D scatter plot projection. It is useful for finding corresponding variables and making further insights about the correspondence. Correspondence analysis can produce rich insights. It answers the question: *how* do selected variables correspond with each other?

Currently correspondence analysis is available in statistical analysis suites that are aimed at a specialized user base. If the process is automated to a comprehensive extent, and delivered as a software service component, correspondence analysis can be a useful tool for non-expert users. Browser-based output visualisations automatically generated from readily available data are an accessible and inexpensive way for researchers

to read their data, regardless of expertise area. The research questions of the thesis follow:

- Is data that is suitable for correspondence analysis commonly enough available?
- Can all statistical assumptions of correspondence analysis be automatically tested?
- Can false interpretations of correspondence analysis be avoided in a service implementation?

In summary, the problem setting of the thesis is the service viability of correspondence analysis.

The application area of exploratory correspondence analysis has been suggested to be “nearly limitless” (Lam 2016). The aim of the thesis work was to utilize correspondence analysis in a practical situation involving a vague research plan, fragmented data and a moderate sample size (reply counts of less than 3322). To demonstrate delivering correspondence analysis as an automated web service, an example software prototype was designed and implemented. The prototype automates statistical tests and the mathematical operation producing visualisations out of source data. It does not automate the very preliminary operations of determining the applicability of correspondence analysis and identifying the data format.

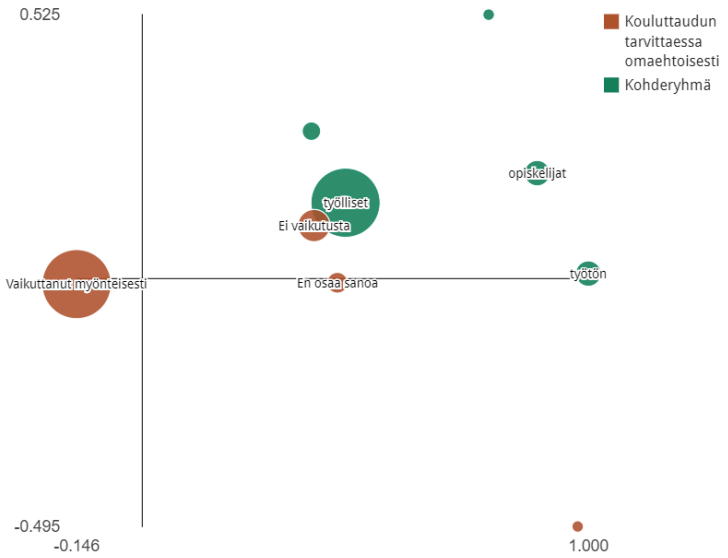

Figure 1. Example correspondence analysis output

The replies to multiple-choice questions in an online survey were used as example data. The theme of the survey was working life in Finland, and the categorical variables it contains include work status, education background, general demographics and opinions. Figure 1 displays an example variable pair: two multiple-choice questions. A JavaScript module (Appendix 1) for evaluating the statistical significance and association strength of correspondence analysis inputs was built in order to measure the quality of example data. An existing open source implementation (Colle 2016) implements matrix operations that convert contingency table data to correspondence analysis visualisations. The visualisations were optimized for interpretability.

By analysing the case example data, it was possible to make insights about, for instance, the correspondence of education level and work status during studies. Statistical tests are essential to data analysis. In a service context, statistical tests can be used to automatically aggregate or manually discover interesting data analysis results. Using statistical tests, the example prototype describes correspondence strength and the reliability of the results.

2 Correspondence analysis

2.1 Multivariate categorical data

The application area of correspondence analysis is multivariate categorical data, a broad classification of data. As opposed to so called quantitative or numerical variables, a categorical variable can take values from a *finite set*. A major share of real world phenomena can be modelled as categorical (nominal, ordinal or limited interval) variables:

- Work status (a nominal variable whose value can be e.g. unemployed, full-time employed, part-time employed, self-employed or retired)
- Likert scale (ordinal (ordered) variable whose value can be disagree strongly, somewhat disagree, indifferent, somewhat agree or agree strongly)
- Income level (a limited interval variable with e.g. 5 levels: <1000€, 1000-2000€, 2000-3000€, 3000-4000€, >4000€)

For comparison, some examples of quantitative (continuous and interval) variables are:

- Distance from one point to another (a continuous variable ranging from 0 to ∞)
- Number of apples in a basket (an interval variable comprising of natural numbers: 0, 1, 2…)

Some real world phenomena can be modelled as both quantitative and categorical data types. In the edge case of interval variables, limiting an interval variable produces a categorical variable, although any kind of interval variable can also be treated as quantitative. Categorical variables' extensive relationship to the real world is why most types of software systems produce some kind of categorical data. Multivariate categorical data refers to data containing multiple categorical variables.

The main problem with multivariate categorical data is its typically high dimensionality. Each outcome of a categorical variable creates a unique dimension that can have a quantified unit value of 0 (value not taken) or 1 (value taken), so a categorical variable with five outcomes, such as work status (studying, full-time employed, part-time employed, unemployed, retired) creates a five-dimensional space. This is because, for instance, work statuses not have any true order, nor can numerical distances between work statuses be measured. High-dimensional geometry arising from categorical variables cannot be visualised *as is*, so making insights out of it is difficult.

Quantitative data is the opposite of categorical data. Categorical techniques like correspondence analysis cannot be used on quantitative data. Quantitative variables comprise of continuous and interval (discrete) variables, sometimes referred to as quantitative data or numerical data. To overcome this limitation on data collection level, the research technique can sometimes be revised. Continuous and interval data can also be converted to categorical data: quantifying a continuum produces intervals, while limiting the minimum and maximum value of an interval variable produces an ordinal categorical variable. In practice, applying categorical data analysis techniques on interval data disregards the ordinality of interval variables. Sometimes, this can be a desired feature of the research technique. If converting or revising the technique is not possible due to practical reasons, various types of correlation analysis can produce insights similar to those produced by correspondence analysis.

A typical way to bypass the dimensionality of categorical variables is to utilize count data, if available. For instance, the counts of people with the five education backgrounds mentioned, can be measured with five separate numerical variables. However, in multivariate statistics, this kind of an approach flattens the taxonomical structure and disregards the complexity of the information that the data contains. Sometimes bypassing the complexity of categorical data is desired, but categorical data analysis techniques can reveal details that quantitative data analysis techniques cannot.

If a dataset contains more than one variable, it is multivariate. Dimensionality is also called Degrees of Freedom – in how many different ways can an observation vary? In multivariate statistics, the space constructed by multiple categorical variables has one dimension that is common to all variables, so its dimensionality is the product of the numbers of categories minus one in each variable. For instance, a person (data entry/observation) who has replied to two (multiple) questions (variables) with 5 answer choices (outcomes) in each question has dimensionality of 16 ($n = (5 - 1) \times (5 - 1)$).

2.2 Process of correspondence analysis

Correspondence analysis explores the correspondence of two or more categorical variables. Dependence and association – the more generic statistical terms that encompass correspondence – are defined as the measure of how much the outcomes of a variable depend on the outcomes of the other variable(s). Correspondence is the dependence of

categorical variables. The more descriptive definition of correspondence varies by application area; examples of variables suitable for correspondence analysis include social status categories, biological taxonomies (Ringrose 1987) and quantum states.

In order to provide an overview of the correspondence of selected variables, correspondence analysis produces an approximate two-dimensional scatter plot projection of the high-dimensional geometry of multivariate categorical data (Lam 2016). Visually, it answers the question: *how* do selected variables correspond to each other? Along with factor analysis, correspondence analysis is a technique central to exploring high-dimensional constructs in categorical data (Lam 2016), without flattening taxonomical structures by converting the categories of categorical variables into series of Boolean variables. Historically, its use cases have been scientific, for instance psychological research, but theoretically its use cases can be found in any situation dealing with categorical data (Lam 2016). Categorical data is also produced by all types of software systems. For these reasons, the service potential of correspondence analysis is subject to research.

The full process of applying the core technique called correspondence analysis is defined here. It involves preliminary steps: 1, 2 and 3, and correspondence analysis itself: steps 4 and 5.

1. Confirming the analysis technique; is the data applicable for correspondence analysis or should another technique be applied?
2. Converting data
    a. Converting between formats
    b. Producing contingency tables
3. Statistical tests for effect size and significance; is there a strong effect and with what chance is it a measurement error?
4. Producing correspondence analysis plots
5. Producing insights from correspondence analysis outputs

To automate the full correspondence analysis process, preliminary steps and the core correspondence analysis operation must be abstracted so that the user does not have to select the technique, to process data or to do any of the core mathematics. Insights can be made through human interpretation or a combination of machine learning (Merz 1997) and Natural Language Generation. Example insights by humans are presented in chapter 5.1.

2.3 Contingency tables

The correspondence of categorical variables is derived from their high-dimensional contingency tables (Yelland 2010), also known as cross-tabulation. In the process description of chapter 2.2, generating contingency tables represents the final sub-step of step 2: converting data. Contingency tables are derived from observation data. This chapter demonstrates the conversion from observation data to contingency tables. For demonstration purposes, the data in this chapter is imaginary (it is not from the case example dataset discussed in 3.3, 4 and 5).

Observation: Person	Variable A: Marital status	Variable B: Number of toes
Jalmari	Category A1: Single	Category B1: 5 toes
Konstantin	Category A2: Married	Category B2: Less than 5 toes
Meredith	Category A3: Widow	Category B3: Over 5 toes
Ayano	Category A1: Single	Category B1: 5 toes
Jamaar	Category A2: Married	Category B3: Over 5 toes
Koppel	Category A3: Widow	Category B2: Less than 5 toes

Table 1. Example bivariate observation data

Contingency tables can be derived from any observation data that is categorical and multivariate. Imaginary observation data with six observations that have two variable values assigned to each, is depicted in Table 1. In the case example dataset, observations refer to the people that answered to the survey, variables refer to questions and categories refer to the answers available in each multiple-choice selection.

	Category B1: 5 Toes	Category B2: Less than 5 toes	Category B3 Over 5 toes:	Var. B sums
Category A1: Single	2	0	0	2
Category A2: Married	0	1	1	2
Category A3: Widow	0	1	1	2
Var. A sums	2	2	2	Total sum = 6

Table 2. Contingency table derived from the example observation data in table 1

A contingency table derived from the observation data of Table 1 is depicted in table 2. The table rows and columns are histograms – observation distributions – of the *co-occurrences* of variable outcomes. Each cell contains observation counts of co-occurrences of variable outcomes. The outcomes for variables A and B are both respectively A1 and B2 in observations #1 and #4, so the observation count of the first cell is 2. The other co-occurrence counts are 0 or 1. The example contingency table can produce some insights, for instance it seems that being single strongly corresponds with having exactly 5 toes. Were the observation counts higher (more about assumptions in next chapter), correspondence analysis could provide a more nuanced overview than the contingency table. In addition, larger contingency tables are more difficult to analyse.

2.4 Assumptions and statistical tests

Some programmatically testable factors that describe the reliability of correspondence analysis are described in this chapter. The reliability of data analysis results is dependent on several factors, some primary ones being effect size, statistical significance and random sampling (Cairo 2016). The additional basic assumptions in correspondence analysis are that the data is multivariate, categorical and unpaired. Paired data refers to nominal variables where one observation does not equal one test subject. In user interface terminology, paired data is checkbox data, as opposed to radio button data.

In data analysis, the total sample size should always be large enough in relation to the measured population. There are several ways of calculating a required research sample size from a given population at a given confidence level. (Cairo 2016) The statistical significance of specifically a categorical contingency table can be evaluated in several ways. Pearson's chi-squared test is commonly used for tables larger than 2 × 2.

Correspondence analysis assumes that the variables examined are independent. Mathematically, this means that the chi-square statistic of the contingency table should not be too high. The threshold value of the chi-squared statistic equal to a Cramer's V (Cramér 1946) value of 0.5. The Cramer's V formula has two attributes: the degrees of freedom and desired confidence level. Cramer's V is a normalized unit variable (a number taking a value from 0 to 1) measuring effect size, or in this case, strength of association. A Cramer's V with a value between >0 and <0.5 means independent but associated. A

value between >0.5 and <1 means dependent or redundant, as in, possibly measuring the same thing. Theoretical values 0 and 1, which are not met in significant samples, respectively represent no association and complete association. If Cramer's V implies redundancy, the plot visualisation can still be generated but it might not produce insights. Similarly, also weak associations can be significant.

Also the individual expected frequencies, as in, the frequencies of a theoretical sample constructed as a part of Pearson's chi-squared test, should be considered. For example, a good sample table with dimensions larger than 2 × 2 has no more than 20% expected frequencies below 5, and no expected frequencies of 0. The contingency table portrayed in 2.3 would not fulfil this assumption or any variation of it. When dealing with this kind of data, the reliability of the chi-square significance test can be improved with a technique called Yates' correction (Yates 1934). In compliance, chi-square test results of data from the case example dataset that do not meet the assumption of high enough expected frequencies turn out unreliable. The R core function chisq.test() outputs a warning when frequencies under 5 are encountered, and indeed, the otherwise complying results of the R implementation and the example service prototype in JavaScript, turn out different when the frequency counts are low. In practice, too low individual expected frequencies can be avoided simply with a large random sample.

2.5 Visualising correspondence

In this chapter, the core dimensionality reduction process used to produce correspondence analysis visualisations is described in geometrical terms. Essentially, visualising correspondence is done by constructing an approximate 2D projection of the high-dimensional complex geometry that arises from cross-tabulating multivariate categorical data. For mathematical formulas and a statistical approach, see Yelland (2010). For a programmatic implementation of correspondence analysis visualisations, see Colle (2016). Optimising the visualisations for interpretability is described in chapter 4.2.

Producing a 2D plot requires 2D data, but bivariate categorical data exceeds two dimensions if at least one of the two variables has more than two possible outcomes. Dimensionality – the number of dimensions – in bivariate categorical data is n = (rows - 1) × (columns - 1), where rows = number of outcomes in variable A and columns = number of outcomes in variable B. Consider the rows and columns of the contingency table as

n-dimensional vectors. The row variable's outcomes can be thought of as vectors in a space with dimensionality equal to the number of columns, and vice versa. These vectors sets have n-dimensional geometries that are visually uninterpretable for humans.

Utilizing the so called singular value decompositions of the both ways of the two-way contingency table, two distance components that most contribute to the overall correspondence can be found. In statistical terms, these components contribute most to the chi-squared statistic. Through identifying them, the high-dimensional distance vectors between the outcomes of each separate variable can be projected in two dimensions with minimal information loss.

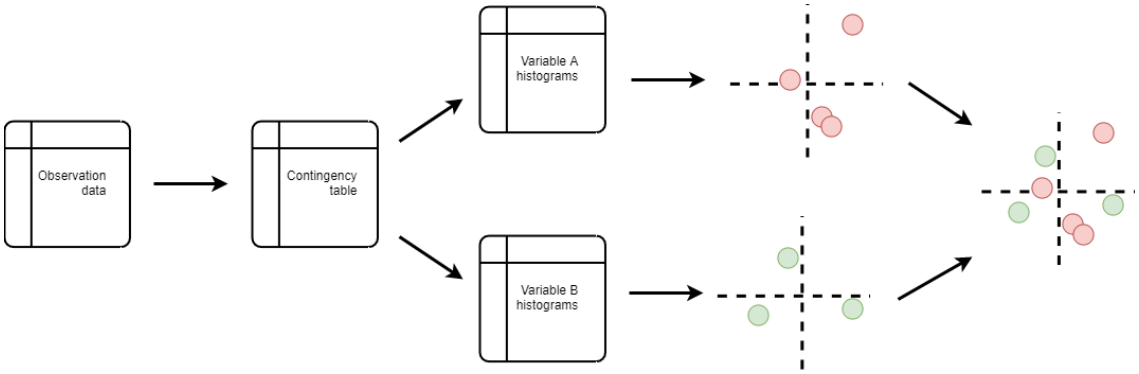

Figure 2. The mathematical procedure of correspondence analysis, visualisation adapted from Lam (2016)

Finally, the abstract axes produced for each plot can be aligned to project the correspondence in a single plot, as illustrated in Figure 2. However, overlaying approximate, differently scaled plots is a further approximation. (Ringrose 1987; Yelland 2010) How these approximations limit interpretation is described in chapter 4.3.2. Variance between implementations of the partially ambiguous correspondence analysis process is demonstrated in chapter 4.3.

3 Service potential of correspondence analysis

3.1 Data analysis and visualisation as a service

Software-as-a-service is a delivery model in which the customers of a software product purchase the right to use a software that is typically hosted on the web. The model relies heavily on automation and a large expected customer base, and it contrasts with expensive software products or manual services that are aimed at a smaller customer base. If a data analysis technique can be reliably automated and it has mainstream demand, it is viable to deliver as a software service. Whether this applies to correspondence analysis, is subject to further research. This thesis concludes that when accompanied by statistical tests, automated correspondence analysis results are reliable but they have room for interpretation errors that could be minimized with annotations and enhancement through complementary data analysis techniques. The service viability of automated correspondence analysis depends on whether complementary techniques are made available, and whether its use cases can be automatically recognized.

Data analysis can be defined as deriving insights from data. An insight is defined as an interesting piece of human-readable information. Insights can be very verbose. Reading data to produce verbal insights is also often laborious and complicated, so the task is typically carried out by professional analysts. If needed, data analysis can be automated to some extent. The extent depends on the techniques used as well as qualities of available data. The full extent includes:

1. Identifying analysis techniques that are relevant to a dataset
2. Preparing the data (conversions and transformations)
3. Applying the analysis techniques
4. Producing insights
5. Communicating the insights

Automated data work can be delivered as software services that allow people to make insights from their data without anyone having to manually apply analysis techniques on it.

To communicate insights, the results of data analysis are often presented through data visualisation, such as maps, charts and plots. Visualisation is, in many cases, a more communicable way of describing data than verbal descriptions. Data analysis insights

are derived from reading numerical data, such as geometry in the case of visualisation. When communicated verbally, some information is always left out. Additionally, when a visualisation is constructed correctly with so called lie factor of 1 (Tufte 1983), it does not produce overstatements or understatements, while that risk is high in verbal communication. Visualisation is in some way a universal way to describe data, although due to cultural-historical reasons or perception disorders, reading visualisations can be difficult or impossible for some people.

Correspondence analysis is a special technique in the sense that it is typically used by data analysts but it presents information visually (Lam 2016). One question posed by the problem setting of this thesis is whether correspondence analysis can be communicable to a mainstream user base including users who are not analysts. Furthermore, if not, can it be made more communicable through graphical and verbal enhancements?

Data analysis can be characterized in many ways. A distinction that is relevant to correspondence analysis and software-as-a-service implementations of analysis techniques is that between Exploratory and Confirmatory data analysis. At its simplest, confirmatory data analysis can refer to the usage of several complementary data analysis techniques that confirm each other's results. Confirmatory data analysis encompasses exploratory data analysis: all data analysis techniques can be confirmatory but only some can be exploratory. Exploratory data analysis comprises specifically of techniques and methods that are best suited for data that not much is known about – data has not produced insights yet, and/or data that has been collected for unknown reasons. Exploratory data analysis is preliminary to confirmatory analysis, the results of which can reliably be published in a scientific or political context.

Although the distinction is not very clear, considering the use cases of exploratory and confirmatory data analysis, namely exploratory techniques have viability when implemented as general-purpose software services. Exploratory techniques can reveal insights with relative ease, and the viability of more accessible implementations is subject to research. On the other hand, confirmatory data analysis cannot be truly confirmatory unless used with a certain level of expertise. Data analysis tools used by experts, such as R or SPSS, already serve the potential user base of confirmatory techniques, so competition would be difficult. Also, making confirmatory data analysis guided or otherwise more accessible might be counterintuitive. Choosing the right approach to confirming

data analysis results is not as simple and automatable, as picking up an explorative technique and seeing what comes up, is. Correspondence analysis is an exploratory technique (Lam 2016).

Graphical methods have been suggested to be inferior to other research methods associated with categorical or “qualitative” data (Sloane 2009). Truly, they are prone to misinterpretation and their ability to produce insights is limited. Additionally, dimensionality reduction and the elimination of outliers – techniques used in visualisation and machine learning – can delete important data along with desired clean-ups. As a counterargument, the service potential of visual techniques for categorical data lies especially in their ability to provide insights from complex data *easily*. For instance, in order to make high-dimensional multivariate categorical data readable for a mainstream audience, producing meaningful correspondence analysis visualisations can be automated. That is, all the way from obtaining data and determining its suitability, to ranking results by their significance and effect strength. Furthermore, reading correspondence analysis outputs is relatively easy (as described in chapters 4.3.2 and 5.1).

A thorough service that automates analysing a variety of data types with appropriate techniques best serves a mainstream user base. The problem that correspondence analysis solves is theoretical: the high dimensionality of multivariate categorical data. For this reason, a mainstream user base cannot be expected to be able to identify suitable data and navigate to a correspondence analysis service if one is made available. Examples of plug-in analytics services that also select appropriate analysis techniques and aggregate interesting results do exist (Qualtrics Inc. 2017, Google Analytics Solutions 2017). However, they often use a proprietary data format (Qualtrics Inc. 2017). Widely used software suites like R and SPSS (Lam 2016) provide means to analyse data in open formats, but they are manual tools aimed at a specialized user base. The market available for a software service that combines the accessibility of commercially oriented analytics services and the variety of formats handled by professional statistical analysis tools is subject to research.

3.2 Visualisation of multivariate categorical data

As described in chapter 2.1, multivariate categorical data is high-dimensional. In both data analysis and visualisation, dimensionality can be bypassed by flattening high-dimensional constructs. However, data analysis techniques that are suitable for categorical data, such as correspondence analysis, can utilize dimensionality instead of bypassing it.

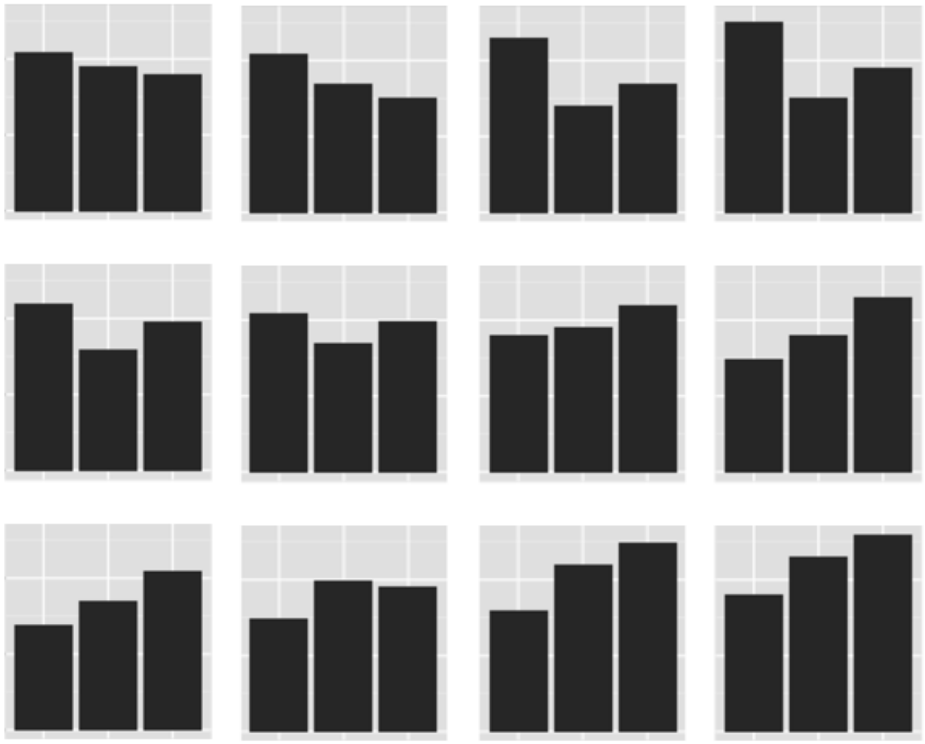

Figure 3. An example of “small multiples” (Tufte 1983) of (univariate) distributions produced in R

Small multiples (Figure 3) refers to a series of some visualisation type produced for multiple variables or observations. One simple approach to visualising multivariate categorical data is to display so called “small multiples” (Tufte 1983) of 1D univariate distributions (Im et al 2013) or high-dimensional contingency tables. A series of 1D visualisations communicates a series of numbers, similarly to a table such as the contingency table of observation data. 2D and 3D visualisations that do not flatten the complex high-dimensional structures in categorical data, such as the 2D plot produced by correspondence

analysis, communicate information through shapes. By utilizing geometrical matrix operations, they can produce more descriptive insights from categorical data, than univariate count data alone can.

The practice of producing utilizable low-dimensional data out of high-dimensional data is called dimensionality reduction. Some classical and contemporary (Oh et al 2001) techniques apply dimensionality reduction to produce 1-4D overviews that are readable for human researchers or optimized for machine learning (Merz 1997) purposes. The core mathematical operation applied in correspondence analysis is a dimensionality reduction algorithm that projects high-dimensional geometry in two dimensions with minimal information loss.

3.3 Case example

The motivation for constructing the example service prototype described in chapter 4 was a dataset containing the results of an online survey conducted by The Finnish Innovation Fund Sitra. The theme of the survey questions is working life in Finland. For the discovery of working life archetypes and the variables that most contribute to the archetypes' existence, the dataset had undergone prior cluster analysis. Correspondence analysis complements cluster analysis (Ringrose 1987), a data analysis technique constructing groups from relationships within the examined observation set. Also, most of the survey questions are categorical data suited for correspondence analysis. In order to evaluate relationships between variable pairs, exploratory correspondence analysis was applied on the case example dataset.

The questions suitable for correspondence analysis in the case example dataset are multiple-choice questions. They measure ordinal categorical variables such as self-evaluations on Likert scale, interval variables such as wealth or nominal variables such as field of study. When using the de facto standard significance test in correspondence analysis – Pearson's chi-squared test for independence (see chapter 2.4) – paired data or so called checkbox data cannot be used. In the case example, this applies to the survey questions for which respondents have been allowed to make multiple selections.

The research question was initially defined as the identification of subgroups and traits in Finnish working life. The data was found to be of good statistical quality, but the nature

of the research question is vague. The data had also been found challenging in a previous analysis. Challenging, fragmented data involving a vague research plan is a good example of, qualitatively, what kind of data can be expected as input data in a general-use service context. This kind of data is suitable for a field case study about the service viability of correspondence analysis. When automated, exploratory methods like correspondence analysis can be used to quickly assess quality of data and to identify variables to apply other analysis techniques on.

user_ID	Q18A_1_1	Q18A_2_1	Q18A_3_1	Q18A_4_1	Q18A_5_1
e439092	5			5	5
e439093				2	1
e439098	1	2			1
e439100				2	2
e439102					1
e439104		1			1
e439107					2
e439108		2	2	1	
e439110			2		
e439127		6		6	
e439134		1			1
e439135		1	2		2
e439138				3	2
e439146		3	2		2
e439148	4			1	

Table 3. An excerpt from the example XLSX data

For demonstration, table 3 portrays an excerpt from the example data. The excerpt demonstrates fragmentation of the data – not all users have answered to all questions. The first column differentiates between users, the total count of which is 5000. The five other columns represent five different multiple-choice questions or categorical variables. The integers in the body area of the table represent answer choices. For instance, Q18A_1_1 represents the question “How likely would you accept unpaid work?”, and the answering options follow Likert scale (integers from 1: not likely to 5: most likely). Based on this, the user on the first row – e439092 – has replied that he is most likely to accept unpaid work.

4 A correspondence analysis service

4.1 Modules of a SaaS implementation

This chapter proposes a modular software architecture for a correspondence analysis implementation. Some modules were implemented in order to construct the example prototype that the later subchapters of chapter 4 follow. To produce a service, the correspondence analysis process defined in chapter 2.2 is extended with user input and usability-related automatable actions.

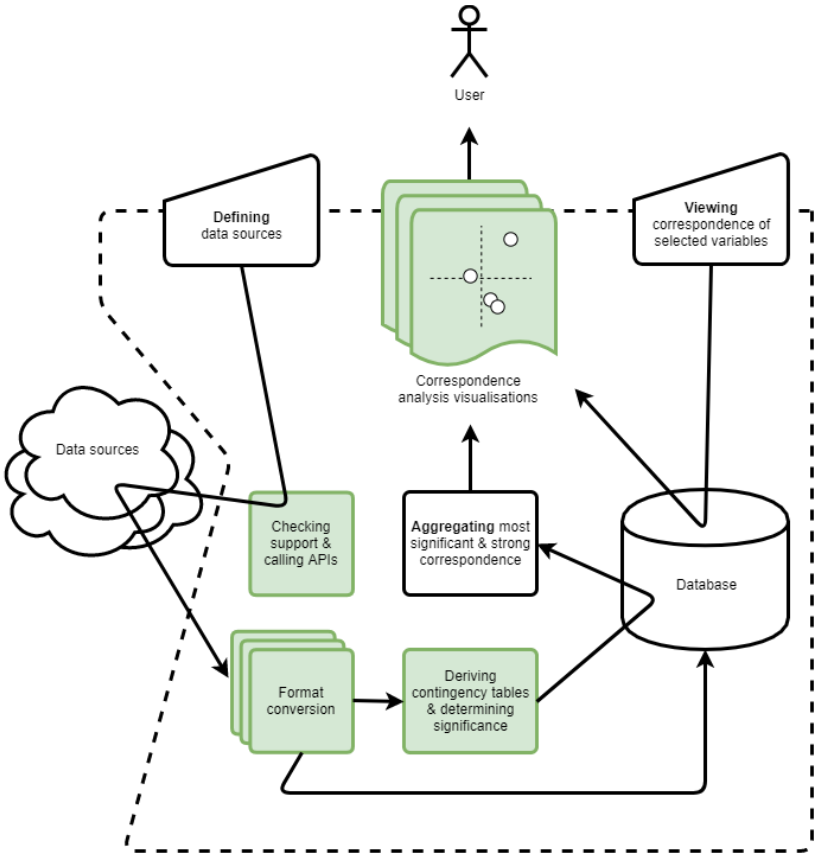

Figure 4. A software-as-a-service implementation of correspondence analysis

Figure 4 is a flowchart of a high-level architecture proposal for a correspondence analysis web service component. The central user actions are 1) defining data sources and 2) viewing correspondence analysis visualisations. Visualisations can be generated from a variety of data sources and delivered through aggregating or user input. Aggregating

refers to delivering visualisations of interesting (significant & strong) variable relationships. The flowchart denotes stages of the correspondence analysis process defined in chapter 2.2.

4.2 Parsing data

The data parsing necessary for a fully automated correspondence analysis implementation includes converting data and performing relevant statistical tests. Conversions include converting between formats and deriving contingency tables from multivariate categorical observation data. The case example observation data was provided in XLSX format of an unrecognized standard or other named convention. In the service prototype that draws from chapter 4.1, all parsing is done in JSON format using the JavaScript language.

In a Software-as-a-service implementation of correspondence analysis, obtaining suitable data, as in, data importing, is followed by constructing contingency tables. Data can be imported from a variety of sources, so it can be saved in a variety of formats. The formats can vary by:

- Encoding language used
- Data structures used
- Structure of metadata, such as legend (the names of data entities).

Each data format requires the development of adapter software. Standards or otherwise popular formats would minimize the repetition of this effort. What comes to viability, it is now very dependent on how much of the market of the relevant data-generating software usage can the analysis service cover. A common business solution to this problem is to develop proprietary formats, but it is beneficial for the growth and life cycle of services that they can be integrated with the software that potential clients already use.

Converting data for the JavaScript prototype starts from producing contingency tables out of observation data as described in chapter 2.3. Statistical tests relevant to correspondence analysis are based on the contingency table. In the example service prototype, a Node.js package connecting contingency table inputs to a comprehensive chi-squared test was constructed (Appendix 1). The probability density function measuring the significance is achieved with an existing Node.js package (Bell 2015). To complement the significance measure, an effect size measure, Cramer's V, was derived from a

Java library (Abeel 2012). In an analytics service, statistical test results can be used to surface strong relationships and significant results. The significance and effect size measures and their relevant threshold values can also be presented to the user as additional information.

The parsing process preliminary to correspondence analysis is relatively computation-intensive. When implemented as a service, such computation-intensive data analysis techniques are best implemented with, for instance, distributed data processing. The case example data is relatively small: 3321 unique pairs of 82 survey questions answered by 5000 people – megabyte-scale. Still, the parsing done for it in JavaScript took over a minute on a decent PC.

4.3 Optimised correspondence analysis visualisation

4.3.1 Constructing a bivariate plot in abstract space

The output of correspondence analysis is a scatter plot visualising the essence of the correspondence of compared variables in approximate 2D space. Reference implementations (Husson et al 2017; Yelland 2010) use a dot to denote of each outcome of the variables. In the example service prototype, a Google Charts Bubble Chart (Google Charts, 2016) visualises the geometry calculated by an open source JavaScript library implementing the matrix operation that is central to correspondence analysis (Colle 2016). The chart is rendered with SVG.

Communicationally this plot differs from a generic scatter plot by presenting the row and column points as spheres of varied size. As a measure of scale, the sum of observations is used. The sum functions as additional information, particularly “mass” in correspondence analysis terminology (Yelland 2010), but must be differentiated from the measure of contribution, which is presented as the distance of the bubble from the origin.

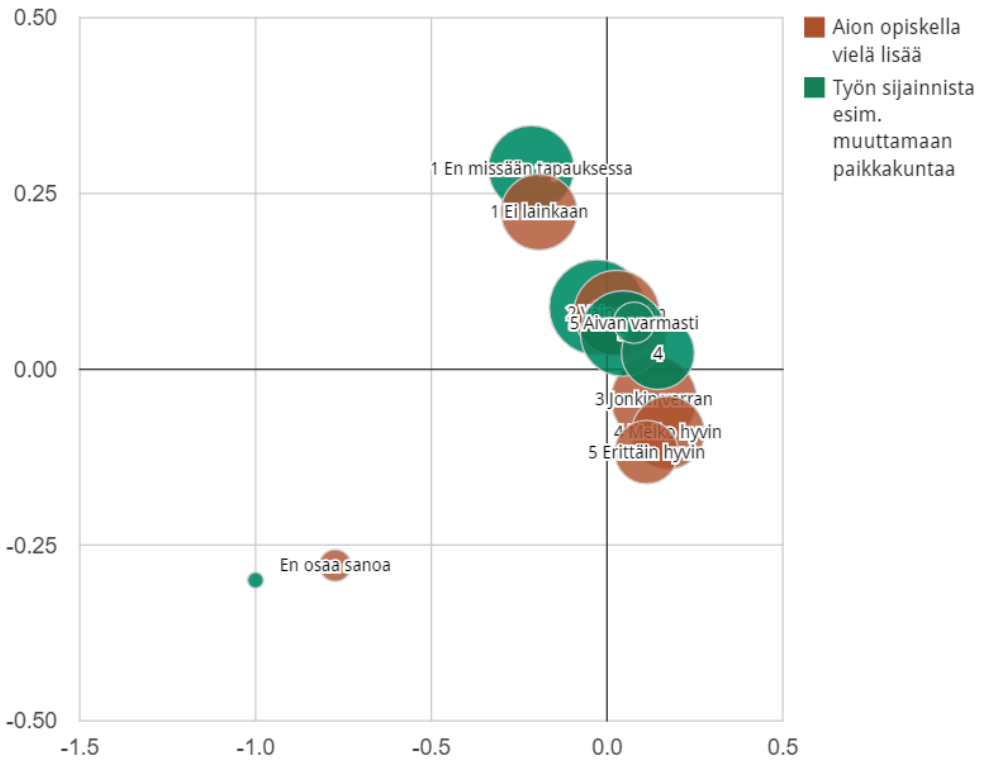

Figure 5. Default output from Google Charts Bubble Chart

The default Bubble Chart plot output by the Google Charts library has many problems in the context of correspondence analysis. Maximising the so called “data-ink” ratio, which refers to eliminating graphical elements that do not communicate relevant information, and including more of those that do, is central in optimizing data graphics (Tufte 1983). Following Tufte’s reasoning, the following changes are made so the chart better communicates correspondence:

- Removing grid lines. The 2D projection is an approximate abstract space, so the grid lines do not communicate *key* information. The origin will still be included as it has a key purpose in interpreting correspondence analysis.

- Optimizing the bounds. Setting the outermost ticks to the points furthest from the origin optimizes the scatter to cover the entire available space. Because the projection is approximate, grid ticks are also redundant. However, outermost coordinate values are left visible as they communicate information: low absolute values imply low association along with the association measure, Cramer's V.
- Removing grid mask. A mask clips the bubbles outside the grid element. The bounds could be adequately extended by adding to the absolute values of the outermost ticks or using outermost ticks with an absolute value slightly over 1, but these solutions would de-optimize the relative visual repulsion from the origin to the bounds. The chart already has plenty margin, allowing the bubbles to overflow, so the grid mask is disabled.
- Confirming the lie factor of the bubbles. Tufte (1983) defines lie factor as the ratio of the size of effect shown in the graphic and the actual size of effect. In this case, the numerical value that the circle denotes should, in relation, equal the area. The online documentation of Google Charts does not state how the size parameter translates to bubble dimensions, so it is assumed the lie factor is 1 (no lie).

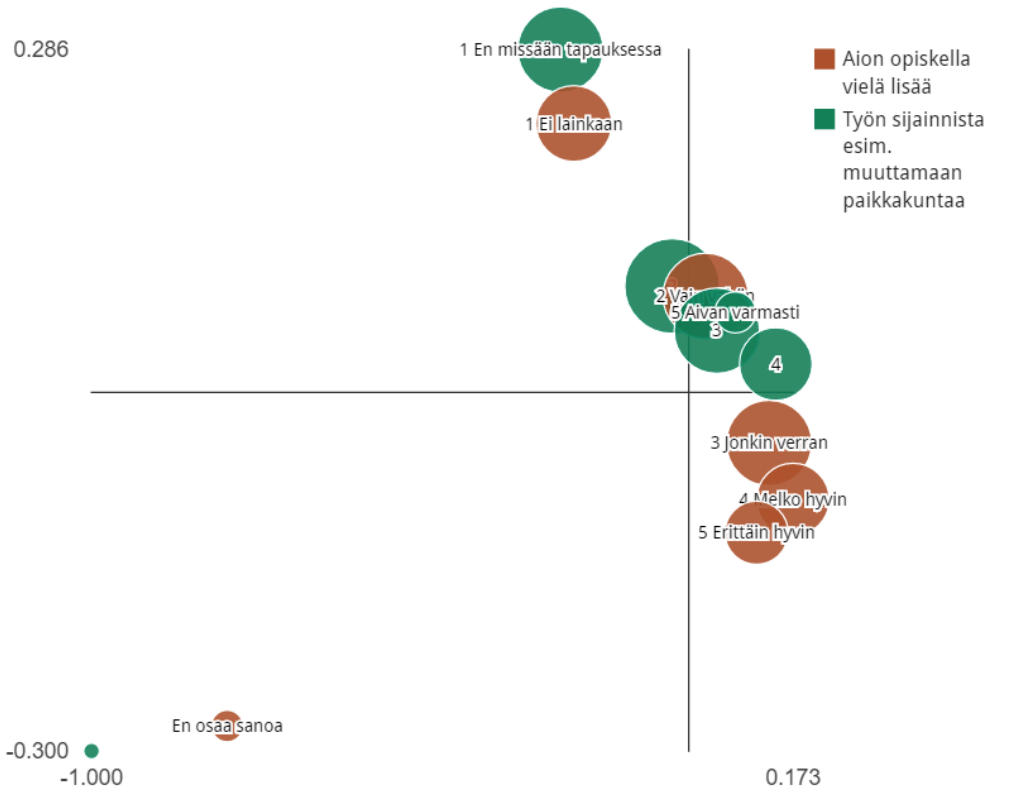

Figure 6. Google Charts Bubble Chart modified for correspondence analysis

Having maximized the data-ink, the graphic communicates the truthful and relevant information more clearly. It is less crammed and less prone to misinterpretation, such as interpreting relative distances as exact. Also, none of the text cuts off. The smaller, less populated column and row bubbles have no title, but in the interactive browser visualisation secondary data such as mathematical metadata and titles of smaller points are presented in the tooltip element.

4.3.2 Interpretation

If the assumptions of correspondence analysis are met and the bivariate plot is constructed right, it can reveal several things about the data, but also lead to misinterpretation. This subchapter describes what special insights the correspondence analysis can provide, what else it describes, and what it does not describe. Let us reflect the presented notes about correspondence analysis interpretation on the following excerpt from the case example data.

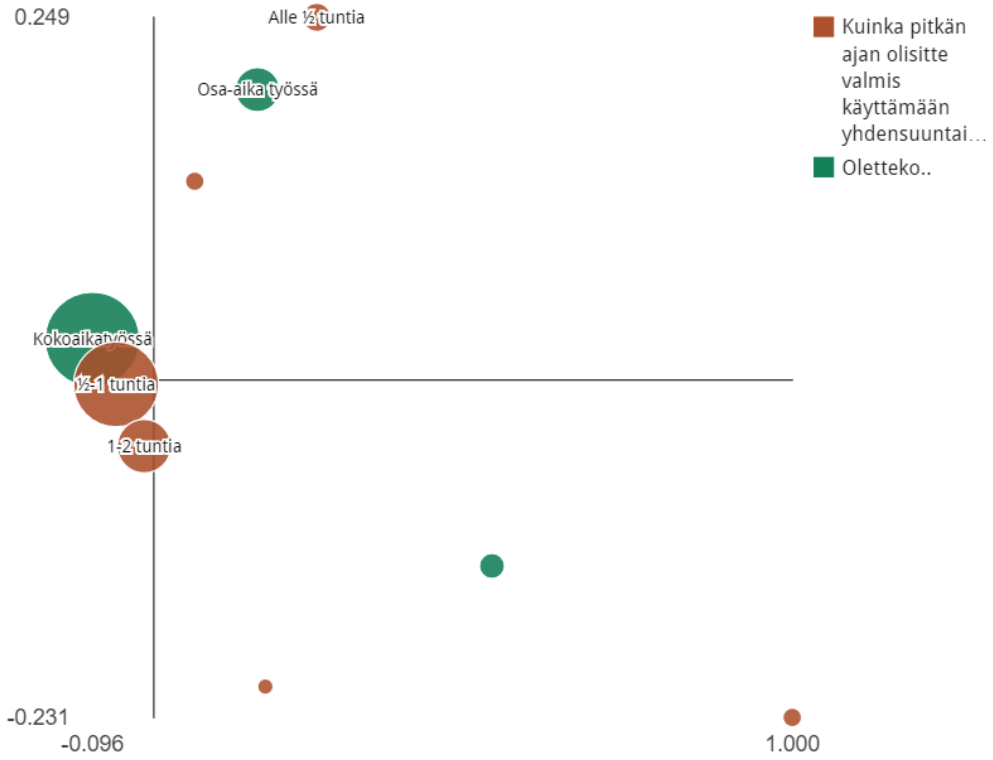

Figure 7. Correspondence analysis result of respondents' preference of maximum commute length and work status (full-time/part-time/both)

The clearest insights provided by the plot are related variable outcomes. Projecting the distributions of the variables is achieved through constructing an approximation of their positions in abstract 2D space. An example result is portrayed in figure 7. Particularly outcomes projected close to the origin are very approximate. For these reasons, only large differences between the outcomes from the same variable can be compared. In addition, plotting each variable in the same plot is done by simply overlaying plots – yet another approximation. To make insights about the relationship between outcomes from different variables, polarising outcomes can be identified with the aid of the axes. (Ringrose 1987; Yelland 2010) Chapter 5.1 demonstrates how the correspondence analysis output varies between different implementations.

The special outcomes of correspondence analysis are identified via “shapes” in the data. For instance, if clusters can be identified through visual inspection or cluster analysis (Ringrose 1987), they imply the existence of a meta-category or a trend among the research sample. Clusters close to the origin may denote a “normal” set of outcomes, but if the association measure of the contingency table signifies very weak association and/or all the outcomes are close to the origin, any result is questionable.

Another type of geometrical phenomenon that arises from dimensionality reduction, is called the horseshoe effect (Diaconis, Goel and Holmes 2008). The name is self-explanatory; horseshoe effect occurs when the points in the plot appear to follow an ordinal scale but also form a curve. This implies that extremes meet, but it can also be accounted to the technique as an error; in different branches of science, the horseshoe effect has been attempted to theorise, or on the other hand, correct. Examples can be found in a variety of fields, such as research of the political left-right spectrum (Diaconis, Goel and Holmes 2008) and rock layers (Ringrose 1987).

The horseshoe effect can be observed in the plot of our case example (Figure 7). Many ordinal variables – categorical variables with order – show up as a curve but interestingly, some nominal variables – unordered categorical variables – also form a line or curve. This may imply that a variable thought to be nominal has an underlying order or ordinality, which becomes visible in the context of the corresponding variable. Ordinality can be also used to convert categorical variables to quantitative variables, enabling correlation analysis.

5 Evaluation

5.1 Case example results

The case example data is of good quality for correspondence analysis per chapter 2.4. The sample size is sufficient in relation to the measured population. Out of the 3321 pairs of 82 categorical survey questions, 1929 have both statistical significance (>95% confidence by Pearson's chi-squared test) and independence (<50% effect size by Cramer's V). In exploratory analysis, also lower significance level can be used; mostly the desired significance level depends on the branch of science.

Cluster analysis and multiple correspondence analysis complement. The multiple correspondence (correspondence of more than two variables) was not evaluated, but the data quality can motivate plotting small variable groups, such as clustering results, in a multiple correspondence analysis plot. Additionally, cluster analysis had been the basis of a prior analysis conducted on the case example data. One of the research questions seeks working life archetypes, while clustering is a good tool for discovering archetypes and other groupings.

Additionally, 61 pairs have both significance and redundancy (Cramer's V > 50%), which implies that each pair of questions measures the same thing. This may be a desired result if survey questions are designed to evaluate response bias, such as, whether respondents understand the questions right. Correspondence analysis is less likely to produce insights from highly redundant questions, but Cramer's V, the dependence pre-test, is useful to evaluate this specific quality.

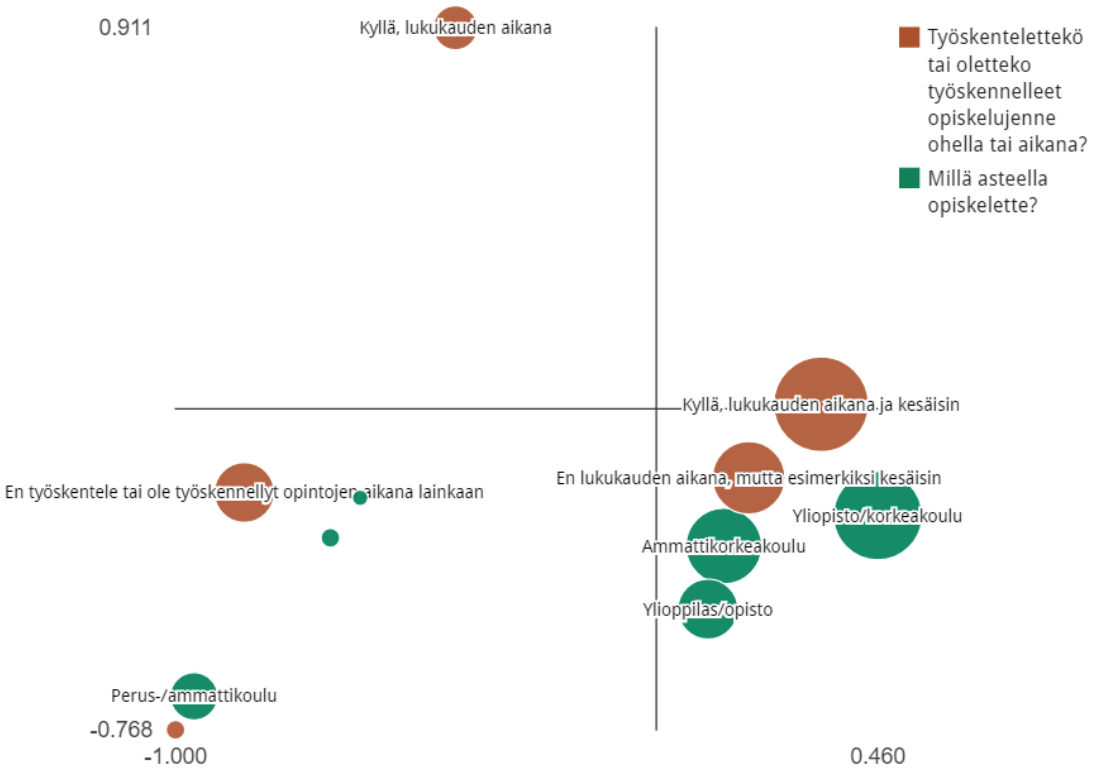

Figure 8. An example of a correspondence analysis result that is statistically significant ($p \longrightarrow 0$), dependent (Cramer's V = 0.21) but not redundant (Cramer's V < 0.5) and capable of producing insights.

Several insights can be made about some of the visualisations produced. One example of such a visualisation is portrayed in figure 8. In the lower left quarter, a trait can be identified – typically, people that have not worked during studies at all have either very low (basic or vocational) or very high (doctorate) education level. In addition, the upper left quarter contains a bubble denoting anomaly occurrences of people who have worked during studies but not during holidays – this implies that it is atypical for people of any education level to have worked only during studies but not during holidays.

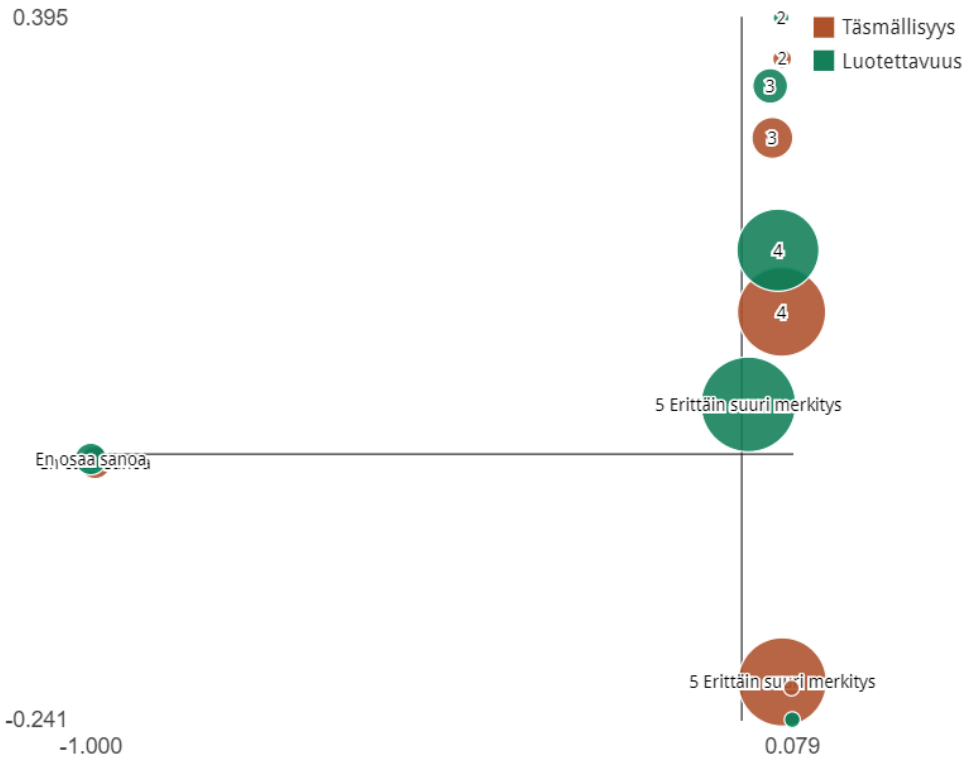

Figure 9. A redundant correspondence analysis result

An example of a poor correspondence analysis result is portrayed in figure 9. Its Cramer's V value is 0.7 (over 0.5), so the relationship of the variables is redundant. This means that with a strong possibility, the variables measure the same thing. In other words, they are not independent.

5.2 Comparison to R output

The main module of the service prototype of chapter 4 is an open source Javascript module implementing the mathematics necessary for a correspondence analysis plot visualisation (Colle 2016). To accompany, a statistical testing module was derived from

an existing Javascript module and a Java library (Abeel 2012). This kind of sources make the system subject to benchmarking, so in this chapter the example prototype's output is compared to output from R packages included in the core packages and other available packages. R is a programming language that is widely used for statistical computing and graphics. Both visualisation output and statistical test output is compared. The source code of the JavaScript module is included in Appendix 1.

The visualisations produced with the JavaScript implementations (Colle, 2016; Google Charts, 2016) used in the example prototype service, and a correspondence analysis function in FactoMineR, an R package aimed for multivariate data analysis (Husson et al. 2017), are presented in the following figures 10 and 11. They are similar, but vary as follows.

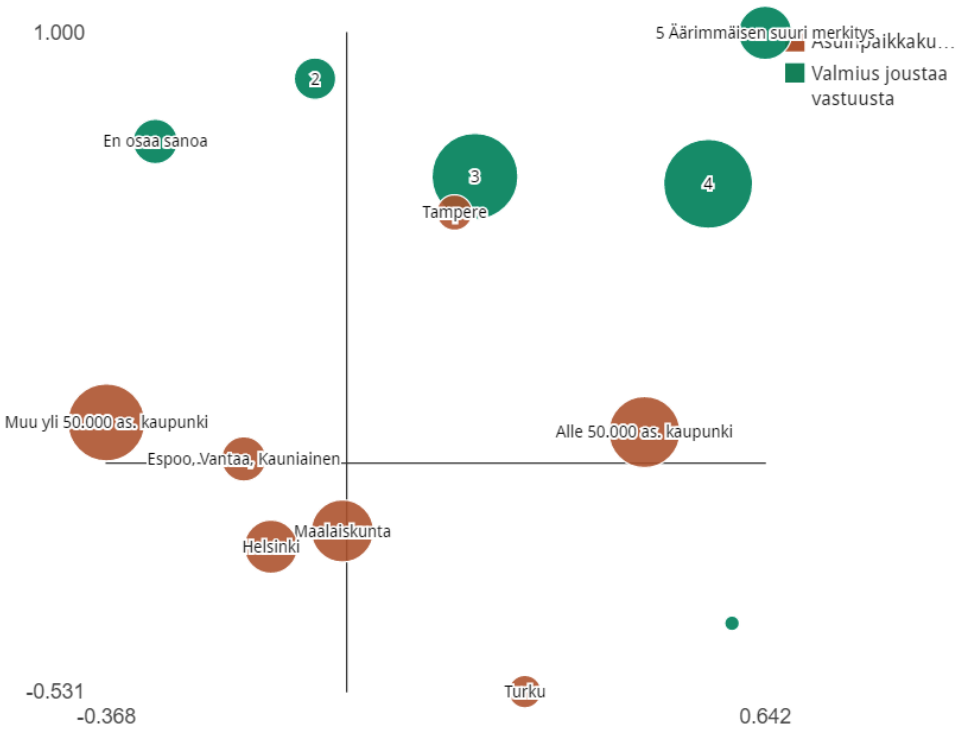

Figure 10. Optimized correspondence analysis output from Google Charts Bubble Chart

A correspondence analysis plot visualisation, optimized according to Tufte's methodology (Tufte 1983), is presented in Figure 10. This 2D projection of the high-dimensional geometry of survey responses is generated with an open source JavaScript implementation (Colle 2016) and Google Charts (Google Charts 2016). In addition to the 2D coordinates, it portrays sample sizes denoted by bubble size.

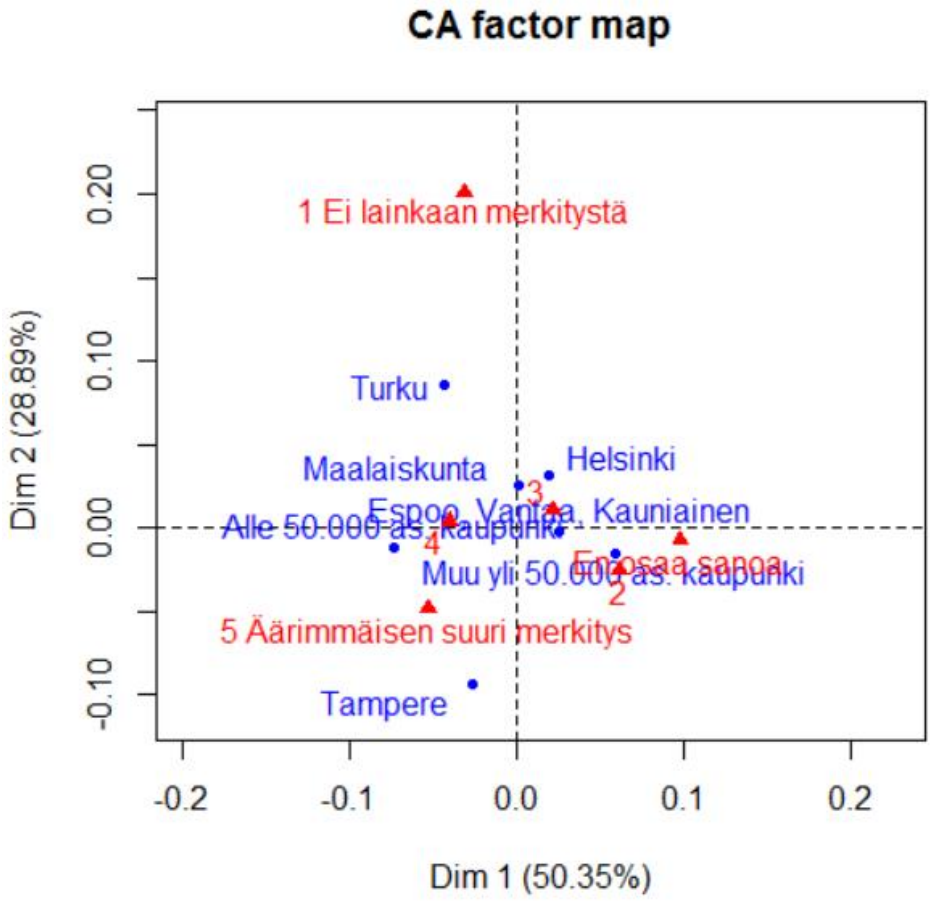

Figure 11. Output of CA() function from the FactoMineR package for R

The plot visualisation generated with an open source R implementation (Husson et al. 2016) is presented in Figure 11. Compared to the example implementation presented in chapter 4, its geometry is similar. This legitimises Colle's (2016) Javascript implementation. The geometry of the two implementations differ by a 180 degree rotation, but the geometries of individual variables appear very similar between implementations. Variable outcomes projected close to the origin vary more than those projected far from the origin.

The coordinates of the correspondence analysis plot have variance between implementations. This demonstrates how comparing the relative distances between outcomes from different variables is erroneous (Ringrose 1987). Due to the variation, interpretation can also differ. In addition, the scaling of the dots used in the JavaScript implementation (Colle 2016) can affect interpretation.

```
Chi-square statistic
          = 23.202996097750173
Pearson's chi-squared test for independence
          = 0.806824162855827 (Not significant)
Cramer's V
          = 0.03046505939449334 (Weak association)
```

Listing 1. Output of the statistical tests performed by Javascript

The example implementation produces verbal descriptions of each figure based on threshold values. The chi-squared statistic is included for comparison. The significance measure is the p-value from Pearson's chi-squared test, and the effect size or dependence strength measure is the Cramer's V.

```
Pearson's Chi-squared test
X-squared = 23.265, df = 30, p-value = 0.8042
```

Listing 2. Output of chisq.test() included in the core packages for R 3.3.2

Comparing the results of R's chisq.test() function and the Node package used in the example prototype, it is evident that they produce similar results with slight variance. This may be due to that R is a language designed for calculations, while Javascript is poor at handling of long decimals. Whether these functions implement Yates' correction (Yates 1934) also contributes to accuracy. In the end, significance depends mainly on the sample size, the sample quality and the research method.

Cramer's V, the preferred effect size measure, is essentially the chi-square statistic shown in the text outputs above, but rescaled to have a value between 0 and 1. R version 3.3.2 released in 2016 does not include Cramer's V in its core packages, so there is no strong reference, but Values of chi-square statistics in R's results and the results of the

example service prototype implementation comply. The Cramer's V formula is very simple, so even taking Javascript's poor decimal handling into account, the Cramer's V value produced is reliable when the assumptions of the chi-square test are met.

5.3 A viable correspondence analysis service

The service viability of correspondence analysis depends on interpretability, reliability and the availability of data. This chapter reflects upon those factors. It should also be noted that automated correspondence analysis software alone has low service viability. Realistically correspondence analysis should be a part of an analytics service that caters to other varieties of data than just multivariate categorical data, and because of its limitations and exploratory nature, it should be complemented with other analysis techniques.

Based on the data provider's feedback, the output of the example service prototype is interpretable and insightful. However, interpreting visual data analysis techniques accurately requires some knowledge of the possibilities and limitations of the technique. Also, classical correspondence analysis plots would benefit from additional information, such as named axes and graphical information. Complementary techniques that can be used to produce verbal and graphical outputs include cluster analysis and Principal Components Analysis (Ringrose 1987). To produce extended overviews, the correspondence analysis visualisation can plot more than just two variables. This is called multiple correspondence analysis. For quantitative data, types of correlation analysis can be implemented in a similar manner to the example prototype.

Implementations of statistical analysis techniques in JavaScript exist. Considering production-quality JavaScript implementations as opposed to the prototype demonstrated in chapter 4, JavaScript's poor decimal handling makes statistical technique implementations in JS subject to scrutiny. Also, the developer community of statistical analysis tools for JavaScript is relatively small to the more scientifically oriented community of the R language. On these terms, also the Java language is a considerable option for back-end analysis calculations. On the other hand, the example prototype was found to produce accurate results when compared to a mature implementation available for R. What comes to computation speed, the example prototype lacks optimization. For instance a horizontally scaled Node.js application can achieve faster computing speed.

The value of a data analysis service depends to some extent on whether users understand what to do with the service. Even though correspondence analysis is relatively usable for e.g. marketing professionals, it can be confused with similar techniques and picking the right technique for the right data type is not an accessible task for someone unfamiliar with data science. In the context of this thesis, this means that viably deploying a correspondence analysis service could be done as a part of a larger software suite. Reviewing the online value proposition of Qualtrics Llc. (2017), one way to viably deploy multivariate analysis techniques as a service is to “automatically run the right statistical tests and surface the strongest relationships” and to “suggest the right visualizations”.

Also adapters should be produced, as discussed in chapter 2.2. Some existing services utilize the benefits of a proprietary data format (Qualtrics Llc. 2017). The market availability of producing correspondence analysis outputs from open formats is subject to research. Some widely used software that produces categorical data suitable for correspondence analysis includes marketing automation software such as Hubspot and Adobe Marketing Cloud, analytics software such as Google Analytics and survey software such as Google Forms.

6 Conclusion

In this thesis correspondence analysis was described, demonstrated and evaluated from the viewpoint of service development. Through a case example, the general viability of implementing correspondence analysis alongside similar data analysis techniques as a part of an automated research service was evaluated. To explore and demonstrate the development process in general, a prototype described in chapter 4 was built by combining existing implementations (Colle 2016; Google Charts 2016) and porting software (Appendix 1). Finally, the prototype was used to analyse example data from a multiple-choice survey.

As described in chapter 5.1, the case example data was found to yield statistically significant results and the implementation designed for interpreting it was able to produce insights. It can be concluded that correspondence analysis is useful in light, applied exploratory research, and the viability of implementing correspondence analysis as a part of an accessible, automated analytics service is subject to further research.

References

Abeel, Thomas. Class ContingencyTables from Java Machine Learning Library [software]. Version 0.1.7. 2012. URL: http://java-ml.sourceforge.net/api/0.1.7/net/sf/javaml/utils/ContingencyTables.html. Accessed 22 April 2017.

Bell, Chip. chi-squared-test [software]. 2015 URL: https://www.npmjs.com/package/chi-squared-test. Accessed 22 April 2017.

Cairo, A. The truthful art: Data, charts, and maps for communication. United States: New Riders Publishing; 2016.

Colle, Pierre. CorrespondenceAnalysis [software]. Version 0.0.5. 2016. URL: https://github.com/piercus/CorrespondenceAnalysis/tree/candidate. Accessed 22 April 2017.

Cramér, Harald. Mathematical Methods of Statistics. Princeton: Princeton University Press; 1946.

Diaconis, P., Goel, S. and Holmes, S. Horseshoes in multidimensional scaling and local kernel methods, The Annals of Applied Statistics 2008;2(3):777–807.

Google Inc. URL: https://www.google.com/analytics/. Google Analytics Solutions [online]. Accessed 24 June 2017.

Google Inc. Google Charts [software]. Version 45. September 12, 2016. URL: https://developers.google.com/chart/

Husson, F., Le, S., Mazet, J. and Josse, J. FactoMineR [software]. Version 1.35. 2017. URL: https://CRAN.R-project.org/package=FactoMineR. Accessed 19 February 2017.

Im, J.-F., McGuffin, M.J. and Leung, R. GPLOM: The generalized plot matrix for Visualizing multidimensional Multivariate data. IEEE Transactions on Visualization and Computer Graphics 2013;19(12):2606–2614

Lam, C. Correspondence analysis: A statistical technique ripe for technical and professional communication researchers. IEEE Transactions on Professional Communication 2016;59(3):299–310

Merz, C.J. Using Correspondence Analysis to Combine Classifiers. Machine Learning 1999;36(1/2):33–58.

Oh, C.-H., Honda, K. and Ichihashi, H. Fuzzy clustering for categorical multivariate data. Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference 2001.

Qualtrics LLC. Meet the Qualtrics Experience Management Platform™ [online]. URL: https://www.qualtrics.com/platform/. Accessed 23 April 2017.

Ringrose, T. Correspondence analysis as an exploratory technique for stratigraphie abundance analysis. Computer and Quantitative Methods in Archaeology 1987:3–14.

Sloane, D. J. Visualizing Qualitative Information. The Qualitative Report 2009;14(3):488–497.

Tufte, E.R. The visual display of quantitative information. 2nd ed. Cheshire, CT: Graphics Press; 1983

Yates, F. Contingency tables involving small numbers and the x^2 test. The Journal of the Royal Statistical Society 1934;1(2):217

Yelland, P. An introduction to correspondence analysis, The Mathematica Journal 2010;12.

JavaScript module: significance measure and effect size for two-way contingency tables

Attached is the source code of the statistical testing module implemented for the case example system. It produces a significance measure (p-value of Pearson's chi-squared test for independence) and an effect size measure (Cramer's V) for contingency table inputs. The module has a dependency to another JavaScript module implementing a generic chi-squared test for 1D vectors (Bell 2015). In the service prototype described in the thesis, the purpose of the module is to evaluate the validity of correspondence analysis results.

```
// Requires NPM package "chi-squared-test" by Chip Bell
var chiSquaredTest = require("chi-squared-test");

var PCSTFI = function(){

    this.degf;
    this.colSums = [];
    this.rowSums = [];
    this.total;
    this.pcst;
    this.contingencyArray = [];
    this.expArray = [];
    this.degfdelta;
    this.cramersV;

}

// Calculate "Degrees of freedom"
PCSTFI.prototype.setDF = function () {
    this.degf=(this.contingencyArray[0].length-1)*(this.contingencyAr-
ray.length-1)
};

// Row sums, column sums, total sum

PCSTFI.prototype.setRowSums = function()
{
    for (var i=0 ; i<this.contingencyArray.length ; i++)
    {
        var s = 1;
```

```
        for (var j=0 ; j<this.contingencyArray[0].length ; j++)
        {
            s += this.contingencyArray[i][j];
        }
        this.rowSums[i]=s;
    }
    return true
}

PCSTFI.prototype.setColSums = function()
{
    for (var j=0 ; j<this.contingencyArray[0].length ; j++)
    {
        var s = 1;
        for (var i=0 ; i<this.contingencyArray.length ; i++)
        {
            s += this.contingencyArray[i][j];
        }
        this.colSums[j]=s;
    }
    return true
}

PCSTFI.prototype.setSampleTotal = function()
{
    var s=0;
    for (var i=0 ; i<this.contingencyArray.length ; i++)
    {
        var x=this.rowSums[i];   // get the next row sum
        s+=x;       // add it to the total
    }
    this.total=Number(s);
}

// Construct a table of "expected frequencies"
PCSTFI.prototype.setExpArray = function()
{
    for (var i=0 ; i<this.contingencyArray.length ; i++)
    {
        this.expArray[i] = [];
        for (var j=0 ; j<this.contingencyArray[i].length ; j++)
        {
            this.expArray[i][j] = (this.rowSums[i] * this.colSums[j])/this.to-
tal;
```

```
        }
    }
}

// Calculate significance measure: Pearson's chi-squared test
PCSTFI.prototype.setPCST = function()
{

    this.pcst = null;

    var flattenedContingencyArray = [].concat.apply([], this.contingencyAr-
ray);
    var flattenedExpArray = [].concat.apply([], this.expArray);

    //reduce this number of degrees of freedom from a generic chi-square test
    this.degfdelta = flattenedContingencyArray.length - this.degf;

    this.pcst = chiSquaredTest(flattenedContingencyArray, flattenedExpArray,
this.degfdelta);

}

// Calculate effect size measure: Cramer's V
PCSTFI.prototype.setCramersV = function(){
    var min = Math.min(this.contingencyArray.length-1, this.contingencyAr-
ray[0].length-1);
    this.cramersV = (min === 0) ? 0 : Math.sqrt(this.pcst.chiSquared/(this.to-
tal*min))
    return true;
}


PCSTFI.prototype.calculate = function(contingencyArray)
{
    this.contingencyArray = contingencyArray;
    this.setDF();
    this.setRowSums();
    this.setColSums();
    this.setSampleTotal();
    this.setExpArray();
    this.setPCST();
    this.setCramersV();
    return true;
}
```

```
var pcstfi = new PCSTFI();

// Set export object to accept the two-dimensional contingency table input
module.exports = function(contingencyArray){
    pcstfi.calculate(contingencyArray);
    return JSON.parse(JSON.stringify(pcstfi));
}
```